



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

*Am*

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/731,690	12/06/2000	Joseph A. Porkka	205964	9318
23460	7590	05/04/2005	EXAMINER	
LEYDIG VOIT & MAYER, LTD TWO PRUDENTIAL PLAZA, SUITE 4900 180 NORTH STETSON AVENUE CHICAGO, IL 60601-6780			ANYA, CHARLES E	
			ART UNIT	PAPER NUMBER
			2194	

DATE MAILED: 05/04/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

## Office Action Summary

Application No.

09/731,690

Applicant(s)

PORKKA, JOSEPH A.

Examiner

Charles E. Anya

Art Unit

2194

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 02 November 2004.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-30 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-30 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date 11/2/04.
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_.

### **DETAILED ACTION**

1. Claims 1-30 are pending in this application.

### **Claim Objections**

2. Claims 2,9 and 15 are objected to under 37 CFR 1.75(c), as being of improper dependent form for failing to further limit the subject matter of a previous claim.

Applicant is required to cancel the claim(s), or amend the claim(s) to place the claim(s) in proper dependent form, or rewrite the claim(s) in independent form.

### ***Claim Rejections - 35 USC § 112***

3. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4. Claims 1-7 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

The following terms lack antecedent basis:

- i. "the first thread" and "the second thread" on lines 9 and 10 of claim 1, line 2 of claim 4 and line 2 of claim 5.

For the purpose of this office action the Examiner would change the terms "the first thread" and "the second thread" to "the first thread of execution" and "the second thread of execution".

- ii. "the method" on line 1 of claims 3-7.

For the purpose of this office action the Examiner would change the term "the method" to "the computer-implemented method".

- iii. "the second thread's queue" on lines 2 and 3 of claim 3 and line 4 of claim 7.

For the purpose of this office action the Examiner would change the term "the second thread's queue" to "the second thread of execution's queue".

- iv. "the first thread's queue" on line 3 of claim 4.

For the purpose of this office action the Examiner would change the term "the first thread's queue" to "the first thread of execution's queue".

### ***Claim Rejections - 35 USC § 103***

- 5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

- 6. **Claims 1-6 and 8-10 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 5,261,097 to Saxon in view of U.S. Pat. No. 5,911,066 to Williams et al.**

- 7. As to claim 1, Saxon teaches a computer-implemented method for passing a message from a first thread of execution in a process to a second thread of execution in

Art Unit: 2194

the process (Col. 3 Ln. 7 – 15, Ln. 60 – 67, Col. 4 Ln. 1 – 28), instantiating a queue for holding messages at the second thread of execution in the process, each queue comprising a reference to a further queue of the same type (Col. 4 Ln. 16 – 28), interpreting a block of source code at the first thread of execution in the process (Col. 3 Ln. 15 – 19, Ln. 60 – 67) and instantiating the message at the first thread of execution in the process (Col. 4 Ln. 16 – 20).

8. Saxon is silent with reference to the method comprising: placing, by the first thread of execution a reference to the message in the queue of the second thread of execution wherein the reference is usable by the second thread of execution to access the message.

9. Williams teaches the method comprising: placing, by the first thread of execution a reference to the message in the queue of the second thread of execution wherein the reference is usable by the second thread of execution to access the message (Col. 17 Ln. 58 – 67).

10. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Williams and Saxon because the teaching of Williams would improve the system of Saxon by providing means for transferring data (Col. 17 Ln. 58 – 67).

11. As to claim 2, see the rejection of claim 1.

12. As to claim 3, Williams teaches the computer-implemented method of claim 1; further comprising: receiving a reference to the second thread of execution's queue and using the reference to the second thread of execution's queue to perform the placing step (Col. 17 Ln. 58 – 67).

13. As to claim 4, Williams teaches the computer-implemented method of claim 1, wherein the first thread has a queue, the method further comprising: passing, to the second thread of execution, a reference to the first thread of execution's queue to allow the second thread of execution to send messages to the first thread of execution (figure 8 (GetData) Col. 8 Ln. 51 – 67, Col. 9 Ln. 1 – 5).

14. As to claim 5, Williams teaches the computer-implemented method of claim 1; further comprising: sending a signal to the second thread of execution to indicate that a message has been sent to the second thread of execution (“...return\_value...” Col. 8 Ln. 46 – 48).

15. As to claim 6, Williams teaches the computer-implemented method of claim 5, wherein the signal is sent via a platform independent object since the invention is practiced using COM (Col. 4 Ln. 31 – 37).

16. As to claim 8, Saxon teaches a method for passing intraprocess messages between scripting threads in a process (Col. 3 Ln. 7 – 15, Col. 3 Ln. 60 – 67, Col. 1 –

Art Unit: 2194

15), the method comprising: creating a first scripting thread of execution (Col. 3 Ln. 60 – 67), creating a first queue for the first scripting thread (Col. 4 Ln. 16 – 28), the first queue comprising a reference to a second queue of the same type as the first queue (Col. 4 Ln. 16 – 28) and creating a second scripting thread of execution (Col. 3 Ln. 60 – 67).

17. Saxon is silent with reference to passing to the second scripting thread, a reference to the first scripting thread's queue for use by the second scripting thread to send messages to the first scripting thread.

18. Williams teaches passing to the second scripting thread, a reference to the first scripting thread's queue for use by the second scripting thread to send messages to the first scripting thread (figure 8 (GetData) Col. 8 Ln. 18 – 67, Col. 9 Ln. 1 – 5).

19. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Williams and Saxon because the teaching of Williams would improve the system of Saxon by providing means for retrieving data (Col. 8 Ln. 18 – 20).

20. As to claim 9, see the rejection of claim 8.

21. As to claim 10, Saxon teaches the method of claim 8; further comprising: creating a queue for the second scripting thread (Col. 4 Ln. 16 – 28), while Williams teaches passing, to the first scripting thread, a reference to the queue of the second scripting

thread for use by the first scripting thread to send messages to the second scripting thread (Col. 9 Ln. 6 – 34).

**22. Claims 14-16,18-26 and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,141,793 to Bryant et al. in view of U.S. Pat. No. 5,261,097 to Saxon et al.**

23. As to claim 14, Bryant teaches a method for compiling a program having a plurality of sections (Col. 6 Ln. 47 – 55), the method comprising: creating a scripting thread for compiling each section (figure 8 Col. 6 Ln. 23 – 46).

24. Bryant is silent with reference to creating a control thread to asynchronously communicate with each of the scripting threads so that commands can be issued from the control thread to the scripting threads in parallel.

25. Saxon teaches creating a control thread to asynchronously communicate with each of the scripting threads so that commands can be issued from the control thread to the scripting threads in parallel (Col. 3 L. 61 – 67, Col. 4 Ln. 1 – 28).

26. It would have been obvious to one of ordinary skill in the art at the time invention was made to combine the teachings of Saxon and Bryant because the teaching of Saxon would improve the system of Bryant by allowing synchronization of thread execution (Col. 4 Ln. 3 – 6).



27. As to claim 15, see the rejection of claim 14.

28. As to claim 16, Saxon teaches the method of claim 14; further comprising: at the control thread, sending updates to a user interface (Col. 4 Ln. 30 – 40) and processing commands from the user interface in parallel with asynchronously sending commands to the scripting threads (Col. 3 Ln. 64 – 67).

29. As claim 18, Bryant teaches a system for compiling a program having a plurality of sections (Col. 6 Ln. 47 – 55), the system comprising: a computer (figures 2/3), a plurality of scripting threads executing on the computer (Server Application 120/Cgi-bin Application 140/Perl Server Process 160 Col. 4 Ln. 45 – 67), wherein each section of the program is compiled under the direction of a scripting thread of the plurality (figures 7/8 Col. 5 Ln. 63 – 67, Col. 6 Ln. 1 – 46) and a control thread executing on the computer for coordinating the activity of the scripting threads by communicating asynchronously with the scripting threads (Perl Server 160 Col. 4 Ln. 51 – 61, figure 8 Col. 23 – 35). Also see the rejection of claim 14.

30. As to claim 19, Bryant teaches the system of claim 18, further comprising: a means for allowing the control thread to communicate asynchronously with the scripting threads (“...pipe...” Col. 4 Ln. 56 – 61, Col. 6 Ln. 23 – 37).

31. As to claim 20, Bryant teaches the system of claim 18; further comprising: a plurality of queues, wherein each queue is associated with a scripting thread of the plurality of scripting threads, and wherein each queue is adapted to receive messages from the control thread (“...next pipe...” Col. 6 Ln. 23 – 37).

32. As to claim 21, Bryant teaches the system of claim 18; further comprising: a means for sending a signal from the control thread to at least one of the plurality of scripting threads to alert the scripting thread whenever a message is sent to the scripting thread (“...receives...” Col. 4 Ln. 56 – 67, figure 9 Col. 6 Ln. 56 – 59).

33. As to claim 22, Bryant teaches the system of claim 18; further comprising: a script engine executing on the computer (figure 4 Col. 4 Ln. 35 – 44), wherein the script engine interprets scripting language commands for each of the plurality of scripting threads (Col. 6 Ln. 39 – 40) and provides a means for sending a signal from the control thread to at least one of the plurality of scripting threads to alert the scripting thread whenever a message is sent to the scripting thread (“...receives...” Col. 4 Ln. 56 – 67, figure 9 Col. 6 Ln. 56 – 59).

34. As to claim 23, Bryant teaches the system of claim 18, wherein the computer is a first computer, the system further comprising: at least one second computer in communication with the first computer, wherein at least one of the scripting threads executes on the second computer (figure 4 Col. 35 – 67).

35. As to claim 24, Bryant teaches the system of claim 23; further comprising: a network link for enabling the first and second computers to communicate with one another (figure 2 Col. 3 Ln. 55 – 65), a means for allowing the scripting thread executing on the second computer to communicate across the network link with the first computer (figure 4 Col. 4 Ln. 35 – 55).

Although Bryant does not teach the control thread as executing on the first computer, one of ordinary skill in the art would know to implement the system to include the control thread as executing on the first computer since the server system of Bryant can reside on any physical machine (Col. 4 Ln. 23 – 26).

36. As to claim 25, Saxon teaches the system of claim 18, further comprising: a user interface, wherein the control thread is operable to update the user interface without having to wait for the scripting threads to act on messages sent to them by the control thread (“...p1..., pn...” Col. 4 Ln. 16 – 57).

37. As to claim 26, Bryant teaches a system for compiling a program having a plurality of sections (Col. 6 Ln. 47 – 55), the system comprising: a server computer (figure 4 Col. 4 Ln. 35 – 55); a control thread executing on the server computer (figure 8 Col. 6 Ln. 23 – 37), a plurality of client computers (figure 4 Col. 4 Ln. 35 – 55), wherein each client computer compiles a section of the plurality of sections (Server Application 120/Cgi-bin Application 140/Perl Server Process 160 Col. 4 Ln. 45 – 67), and wherein

the client computers are in communication with the server computer, and a plurality of scripting threads executing on the server computer (figure 4 Col. 4 Ln. 35 – 55, Col. 6 Ln. 23 – 55), wherein each scripting thread directs the compiling activity of a client computer of the plurality of client computers (figure 4 Col. 4 Ln. 35 – 55), and wherein the control thread sends messages asynchronously to each of the plurality of scripting threads to coordinate their activities (figure 8 Col. 6 Ln. 23 – 37). Also see the rejection claim 14.

38. As to claim 27, Bryant teaches the system of claim 26, wherein the control thread sends messages asynchronously to each of the plurality of scripting threads to coordinate their activities, thereby resolving interdependencies among different sections of the program that are being compiled (“...pipe...” Col. 4 Ln. 56 – 67).

**39. Claims 7,11-13 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 5,261,097 to Saxon in view of U.S. Pat. No. 5,911,066 to Williams et al. as applied to claim 1 above, and further in view of U.S. Pat. 6,728,722 B1 to Shaylor.**

40. As to claim 7, Saxon as modified by Williams is silent with reference to the method of claim 1; further comprising: defining a message object for holding the message; and inserting the message into the message object, wherein the reference placed in the second thread of execution's queue is a reference to the message object.

Art Unit: 2194

41. Shaylor teaches the method of claim 1; further comprising: defining a message object for holding the message, and inserting the message into the message object (Col. 18 Ln. 26 – 43), wherein the reference placed in the second thread of execution's queue is a reference to the message object (“...pointer...” Col. 13 Ln. 57 – 67, Col. 14 Ln. 1 – 3).

42. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of Shaylor, Saxon and Williams because the teaching of Shaylor would improve the system of Saxon as modified by Williams by reducing the amount data to be transferred in a message passing system (Shaylor Col. 13 Ln. 63 – 67).

43. As to claim 11, see the rejection of claim 7.

44. As to claim 12, Williams teaches the method of claim 11; further comprising: sending a signal from the first scripting thread to the second scripting thread to indicate to the second scripting thread that a new message has been sent to the second scripting thread (“...return\_value...” Col. 29 – 32).

45. As to claim 13, Shaylor teaches the method of claim 11, wherein in response to the message further comprising: inserting a flag in the message object to indicate that it is being responded to and placing a reference to the message object in the queue of the first scripting thread (Col. 17 Ln. 54 – 62).

**46. Claim 17 is rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,141,793 to Bryant et al. in view of U.S. Pat. No. 5,261,097 to Saxon et al. as applied to claim 14 above, and further in view of U.S. Pat. 6,728,722 B1 to Shaylor.**

47. As to claim 17, Bryant as modified by Saxon is silent with reference to the method of claim 14, further comprising: creating a queue for the control thread and passing, to at least one of the scripting threads, a reference to the control thread's queue for use by the scripting thread to send messages to the control thread.

48. Shaylor teaches the method of claim 14, further comprising: creating a queue for the control thread (Col. 18 Ln. 26 – 43) and passing, to at least one of the scripting threads, a reference to the control thread's queue for use by the scripting thread to send messages to the control thread (“...pointer...” Col. 13 Ln. 57 – 67, Col. 14 Ln. 1 – 3).

49. It would have been obvious to one of ordinary skill in the art at the invention was made to combine the teachings of Shaylor, Bryant and Saxon because the teaching of Shaylor would improve the system of Bryant as modified by Saxon by reducing the amount data to be transferred in a message passing system (Shaylor Col. 13 Ln. 63 – 67).

**50. Claims 28,29 and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,141,793 to Bryant et al. in view of U.S. Pat. No.**

**5,261,097 to Saxon et al. as applied to claim 26 above, and further in view of U.S. Pat. 5,911,066 to Williams et al.**

51. As to claim 28, Bryant teaches the system of claim 26; further comprising: a one or more control thread queues associated with the control thread (Col. 4 Ln. 56 – 62).

52. Bryant as modified by Saxon is silent with to reference to a plurality of scripting thread queues, wherein each scripting thread queue is associated with a scripting thread of the plurality of scripting threads, and wherein the control thread has a reference to each scripting thread queue, and wherein each scripting thread has a reference to at least one control thread queue that is associated with the scripting thread, thereby enabling the control thread to put one or more of the messages in each scripting thread queue and each scripting thread to put response messages in the associated queue of the control thread.

53. Williams teaches a plurality of scripting thread queues (“...media...” Col. 8 Ln. 35 – 67), wherein each scripting thread queue is associated with a scripting thread of the plurality of scripting threads (“...client...” Col. 8 Ln. 35 – 67), and wherein the control thread has a reference to each scripting thread queue (“...pformatetc parameter...” Col. 8 Ln. 35 – 40), and wherein each scripting thread has a reference to at least one control thread queue that is associated with the scripting thread, thereby enabling the control thread to put one or more of the messages in each scripting thread queue and each scripting thread to put response messages in the associated queue of the control thread (“...GetData.../...GetDataHere...” Col. 35 – 67, Col. 9 Ln. 1 – 34).

54. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Bryant, Saxon and Williams because the teaching of Williams would improve the system of Bryant and Saxon by providing means for retrieving and transferring data.

55. As to claim 29, Bryant teaches the system of claim 26; further comprising: at least one script stored on the server computer, wherein the script contains instructions for directing the compilation of the program (Server Application 120/Cgi-bin Application 140 Col. 4 Ln. 45 – 67) and a script engine executing on the server computer to interpret the script, the script engine having an inter-thread signaling mechanism (Perl Server Process 160 Col. 4 Ln. 45 – 67), wherein the control thread uses signaling mechanism to alert a scripting thread of the plurality of scripting threads whenever the control thread has sent a message to the scripting thread (figure 8 Col. 6 Ln. 23 – 67).

56. As to claim 30, Bryant teaches the system of claim 26, wherein the control thread sends messages asynchronously to each of the plurality of scripting threads to coordinate their activities, thereby resolving interdependencies among different sections of the program that are being compiled, the system further comprising: a plurality of control thread queues associated with the control thread; a plurality of scripting thread queues, wherein each scripting thread queue is associated with a scripting thread of the plurality of scripting threads (“...pipes...” Col. 6 Ln. 23 – 38), and at least one script stored on the server computer, wherein the script contains instructions for directing the



compilation of the program (Server Application 120/Cgi-bin Application 140 Col. 4 Ln. 45 – 67) and a script engine executing on the server computer to interpret the script, the script engine having an inter-thread signaling mechanism (Perl Server Process 160 Col. 4 Ln. 45 – 67), wherein the control thread uses signaling mechanism to alert a scripting thread of the plurality of scripting threads whenever the control thread has sent a message to the scripting thread (figure 8 Col. 6 Ln. 23 – 67), while Williams teaches the control thread having a reference to each scripting thread queue (“...pformatetc parameter...” Col. 8 Ln. 35 – 40) and wherein each scripting thread has a reference to a corresponding control thread queue of the plurality of control thread queues, thereby enabling the control thread to put one or more of the messages in each scripting thread queue and each scripting thread to put response messages in its corresponding control thread queue (“...GetData.../...GetDataHere...” Col. 35 – 67, Col. 9 Ln. 1 – 34). See the rejection of claim 28.

### ***Response to Arguments***

57. Applicant's arguments filed 11/2/04 have been fully considered but they are not persuasive.

Applicant argues in substance that (1) none of the cited prior art references teach each queue comprising a reference to a further queue of the same type, (2) the Williams prior art reference does not teach the word “thread”, “queue” and “message” and (3) none of the cited prior art references teach a program that is being compiled to be independent of the script executing the compilation.

Examiner respectively traverses Applicant's argument:

As to point (1), the parameters "p1, ..., pn" of Saxon prior art reference (Col. 4 Ln. 16 – 20) are queues used as input or output for passing information between threads and these queues are identical or of the same type.

As to point (2), Applicant's argument is negated by the disclosure of claim 1 of the Williams prior art reference which reads as follows: "A method for transferring data between a source process and a destination process in a data processing system having a plurality of connection mechanisms for establishing logical connections between the source process and the destination process, the method comprising the steps of: providing to the source process and to the destination process a uniform data transfer module for transferring the data from the source process to the destination process; establishing a logical connection between the source process and the destination process utilizing one of the plurality of connection mechanisms; and transferring the data from the source process to the destination process utilizing the uniform data transfer module, wherein the uniform data transfer module transfers the data in a manner that is independent of the connection mechanism used to establish the logical connection such that the uniform data transfer module operates in the same manner to transfer the data regardless of which of the plurality of connection mechanisms is used to establish the logical connection".

As to point (3), referring Applicant to page 7, lines 1-2 of the Applicant's specification which discloses that in order to execute a thread the script engine interprets and acts upon a block of script. Since a thread is a portion of a program that

can run independently of and concurrently with other portions of the program, the thread whose execution is an interpretation of a block script is independent of the program.

### ***Conclusion***

58. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the mailing date of this final action.

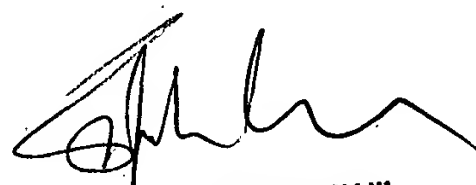
Any inquiry concerning this communication or earlier communications from the examiner should be directed to Charles E. Anya whose telephone number is (571) 272-3757. The examiner can normally be reached on M-F (8:30-6:00) First Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, An Meng-Ai can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Charles E Anya  
Examiner  
Art Unit 2194

cea.



ST. JOHN COURTENAY III  
PRIMARY EXAMINER